

ICT

2

Lesson preparation book

prep2

Second term



Teacher's Biography

Name:

School:

The educational administration:

Qualification:

Teaching Subject:

Comprehensive School:

The school to which he is delegated:

Date of appointment:

The job is on the staff:

Teacher Code:

Mobile Number:

Teacher

Supervisor

School Principal

Daily class schedule

Session Day	First	Second	Third	Fourth	Fifth	Sixth	Seventh	Eightieth	Ninth
Saturday									
Sunday									
Monday									
Tuesday									
Wednesday									
Thursday									

session Day	First	Second	Third	Fourth	Fifth	Sixth	Seventh	Eightieth	Ninth
Saturday									
Sunday									
Monday									
Tuesday									
Wednesday									
Thursday									

Teacher

Supervisor

School Principal

General Objectives of Information and Communication technology (ICT) – second preparatory
Grade, Second Semester, Year 2025/2026

By the end of the second semester, the student should be able to:

✍✍✍ **Understand** the concept of artificial intelligence and its impact on our lives.

✍✍✍ **Understand** the applications of artificial intelligence.

✍✍✍ **Define** the concept of an artificial neural network.

✍✍✍ **Recognize** the importance of programming languages.

✍✍✍ **Explain** the types of loops and functions in Python.

✍✍✍ **Learn** how to use Microsoft Excel.

✍✍✍ **Acquire** proficiency in working with programming languages.

✍✍✍ **Explain** the concept of the Python programming language.

✍✍✍ **Use** robotics and sensor technologies.

✍✍✍ **Learn** the fundamentals of the Python programming language.

Teacher

Supervisor

School Principal

Specific Objectives of the Information and Communication Technology (ICT) Subject

Second preparatory Grade, Second Semester Year 2025/2026

By the end of the second semester, the student should be able to:

- **Distinguish** between the workings of traditional programs and artificial intelligence.
- **Understand** the historical development of artificial intelligence.
- **State** the importance of artificial intelligence in solving problems in our lives.
- **Compare** human thinking and machine thinking.
- **List** some examples of applications of artificial intelligence in our lives.
- **Suggest** some future applications of artificial intelligence.
- **Understand** the ethical responsibility in the uses of artificial intelligence.
- **Suggest** as many ideas as possible for using artificial intelligence technology in responsible ways.
- **Suggest** some simple projects that employ artificial intelligence responsibly.
- **Explain** the concept of artificial neurons.
- **Identify** the components of a neuron and its role in artificial intelligence.
- **Understands** how to build and operate an artificial neural network.
- **Recognizes** the concept of artificial neural networks.
- **Explains** the components of an artificial neural network.
- **Lists** as many ideas as possible for neural network applications in our lives.
- **Understands** the advantages of Python as a suitable language for data analysis.
- **Recognizes** the basic concepts in data analysis (arithmetic mean, maximum, and minimum values).

- **Lists** as many ideas as possible for neural network applications in our lives.
- **Knows** how to import an Excel file containing student grades and view the underlying data.
- **Performs** some calculations (calculating averages, finding the highest and lowest scores in each subject) to determine the number of successful students.
- **Analyzes** the relationships between scores in different subjects through results and graphs.
- **Understands** the concept of conditional statements and how to use them for decision-making in programs.
- **Explains** how to use loops to repeat programming commands effectively.
- **Applies** some examples of functions and how to create and call them to organize code.
- **Understands** the concept of loops and functions.
- **Explains** the while loop for conditional repetition.
- **Writes** simple code for loops and functions.
- **Understands** data structures in Python and why we use them.
- **Compares** the uses of List, Tuple, and Dictionary.
- **Writes** simple code using data structures in Python.
- **Uses** Python functions to convert letters to numbers and vice versa.
- **Performs** some operations (encrypting/decrypting a single character by adding and subtracting a constant).
- **Explains** the importance of encryption. In data and information protection.
- **Explains** the concept/function of bit-level operations.
- **Uses** the XOR operation to encrypt/decrypt simple messages.
- **Writes** a simple Python program to encrypt/decrypt text with a secret key

Teacher

Supervisor

School Principal

The Study Plan for the Information and Communication Technology (ICT) Curriculum for Second preparatory, second semester – Academic Year 2025/2026

Month	week	Date	Topic	Tests & Evaluation	notes
			Unit Three: Artificial Intelligence		
February 2026	1	7/2/2026	Lesson One: The Historical Development of Artificial Intelligence	According to Resolution 136, a weekly test is conducted.	➤ Holidays, official vacations, and exam dates must be taken into consideration. ➤ Eid al-Fitr begins on Saturday, March 21, 2026. The first monthly exam will be held from the 25/3/2026 to the 3rd of April 2026
	2	14/2/2026	Lesson Two: How Machines Think and Change Our World Options?		
	3	21/2/2026	Lesson Three: Artificial Intelligence Is All Around Us		
	4	28/2/2026	Lesson Four: The Artificial Neural Cell		
March 2026	5	7/3/2026	Lesson Five: The Role of the Neural Cell in Artificial Intelligence		
	6	14/3/2026	Lesson Six: Introduction to Data Analysis in Python		
			Unit Four: python programming language		
	7	21/3/2026	Lesson One: practical application of data analysis from an Excel file Lives		
May 2026	8	28/3/2026	Lesson Two: conditional statements in Python Applications		
	9	4/4/2026	Lesson Three: loops and functions		
	10	11/4/2026	Lesson Four: data structures or collection		➤ Sinai Liberation Day, Saturday 24/4 /2026 ➤ Labor Day, Friday 5/1/2026 ➤ Second Monthly Exam from 26/4/ to 30/4/2026
	11	18/5/2026	Lesson Five: an introduction to cryptography using Python		
	12	25/4/2026	Lesson Six: Python cryptography: how to turn messages into secrets using the XOR operator)		
	13	2/5/2026	Project + General Review		The second monthly meeting for parents will held on 7/5/2026

Teacher

Supervisor

School Principal

Grade	Class	Date	Period	Unit	Lesson	Lesson Title
2prep.				Three	One	Unit Three: Artificial Intelligence The historical development of artificial intelligence

Introduction / How have artificial intelligence technologies evolved in our lives?

Lesson Objectives / -To differentiate between the workings of traditional programs and artificial intelligence.

-To understand the historical development of artificial intelligence.

-To state the importance of artificial intelligence in solving problems in our lives.

Skills, values and issues covered in the lesson	Teaching Aids	Assistive technology for integration students if available	Teaching strategies	Subject		Educational activities	Formative assessment	Evaluation	
Learning to Be: Developing Critical Thinking . Work Values: Understanding the Impact of Technology on Society . Globalization Issues: Keeping Pace with Technological Development Throughout the Ages.	Interactive board– Textbook – Introductory Video		☐ Brainstorming ☐ Dialogue and discussion ☐ Cooperative Learning	Key milestones that shaped our current reality: First: Humanity's dream of a thinking, self-operating machine: Long before the invention of computers, humans dreamed of machines capable of thinking and acting like them. Ancient civilizations witnessed efforts that can be considered the initial roots of mechanical thinking, manifested in mythology, philosophy, and early mechanics. Second: The modern era of artificial intelligence:			In collaboration with your classmates, think of at least three examples to determine whether the messages you send to another party are from a machine or a human.	Using PowerPoint, design a poster illustrating the most important applications of artificial intelligence.	Solve the questions in the textbook.
				A - Alan Turing, the father of artificial intelligence, 1950 (the beginning of the emergence of the term)					
				B - Dartmouth Conference, Official Birth 1956					
				C - Years of enthusiasm and great hopes 1956 - 1970					
				D- Expert Systems (1970-1980)					
				E - The Winter of Artificial Intelligence 1980 - 1990					
				F- Renaissance era 1990 - 2010					
				G - The Modern Revolution (2010 - Now)					
Weekly Assessment		Put a checkmark (✓) next to the correct statement and an X (x) next to the following: Artificial intelligence is a branch of computer science. ()							

Teacher's Signature

Grade	Class	Date	Period	Unit	Lesson	Lesson Title
2prep.				Three	four	The Artificial Neural Cell .

Introduction / How do artificial intelligence technologies work? Explain with examples.

Lesson Objectives / - To explain the concept of an artificial neuron.

-To identify the components of a neuron and their role in artificial intelligence.

-To deduce how an artificial neuron is built and operates.

Skills, values and issues covered in the lesson		Teaching Aids	Assistive technology for integration students if available	Teaching strategies	Subject				Educational activities	Formative assessment	Evaluation																					
Learn to be: Increasing technological awareness . Work values: Collaboration. Globalization issues: Facing global challenges through digital transformation.		Interactive board– Textbook – Introductory Video		☐Brainstorming ☐Dialogue and discussion ☐Cooperative Learning	Artificial neuron: A small unit in a computer that attempts to mimic human thought processes. Components of a neuron: <table><tr><th>1. Inputs:</th><th>2. Weights</th><th>3. Bias</th><th>4. Activation Function</th></tr><tr><td>This is the information that enters the cell. For example: Image - Sound - Number.</td><td>These are numbers that help the cell determine the importance of each part of information.</td><td>It is a number added to the total to adjust the cell's operation and make it more flexible. It allows results to be given even when the input is zero.</td><td>It makes the final decision: Does the cell send a signal? Or does it remain silent?</td></tr></table> 4. Activation Function <table><tr><th>Name</th><th>Function</th><th>Practical example (see steps on page 29 of the book)</th></tr><tr><td>Boundary</td><td>Gives (0 or 1) only.</td><td>A number added to the total to adjust the cell's operation and make it more flexible. It allows results to be given even when the input is zero.</td></tr><tr><td>Probability</td><td>Gives a percentage or probability between 0 and 1</td><td>Probability of this being a cat = 0.85 (85%)</td></tr><tr><td>Value Activation</td><td>. Passes only positive values.</td><td>ore very faint sounds in the microphone</td></tr></table>				1. Inputs:	2. Weights	3. Bias	4. Activation Function	This is the information that enters the cell. For example: Image - Sound - Number.	These are numbers that help the cell determine the importance of each part of information.	It is a number added to the total to adjust the cell's operation and make it more flexible. It allows results to be given even when the input is zero.	It makes the final decision: Does the cell send a signal? Or does it remain silent?	Name	Function	Practical example (see steps on page 29 of the book)	Boundary	Gives (0 or 1) only.	A number added to the total to adjust the cell's operation and make it more flexible. It allows results to be given even when the input is zero.	Probability	Gives a percentage or probability between 0 and 1	Probability of this being a cat = 0.85 (85%)	Value Activation	. Passes only positive values.	ore very faint sounds in the microphone		In collaboration with your classmates, think of at least three examples to determine whether the messages you send to another party are from a machine or a human.	Using PowerPoint, design a poster illustrating the most important applications of artificial intelligence.	Solve the questions in the textbook.
					1. Inputs:	2. Weights	3. Bias	4. Activation Function																								
					This is the information that enters the cell. For example: Image - Sound - Number.	These are numbers that help the cell determine the importance of each part of information.	It is a number added to the total to adjust the cell's operation and make it more flexible. It allows results to be given even when the input is zero.	It makes the final decision: Does the cell send a signal? Or does it remain silent?																								
					Name	Function	Practical example (see steps on page 29 of the book)																									
					Boundary	Gives (0 or 1) only.	A number added to the total to adjust the cell's operation and make it more flexible. It allows results to be given even when the input is zero.																									
Probability	Gives a percentage or probability between 0 and 1	Probability of this being a cat = 0.85 (85%)																														
Value Activation	. Passes only positive values.	ore very faint sounds in the microphone																														
Weekly Assessment		• Mark (✓) next to the correct statement and (×) next to the following: Bias helps the cell to produce results even in the absence of input. ()				Teacher's Signature																										

Grade	Class	Date	Period	Unit	Lesson	Lesson Title
2prep.				Three	six	Introduction to Data Analysis in Python

Introduction / How can data be analyzed using Python?

- Lesson Objectives /**
- To understand the advantages of Python as a suitable language for data analysis.
 - To learn the basic concepts in data analysis (arithmetic mean, maximum and minimum values).
 - To list as many ideas as possible for the applications of neural networks in our lives.

Learn to be: Technological awareness Work values: Collaboration Globalization issues: Facing global challenges through digital transformation		Skills, values and issues covered in the lesson	Teaching Aids	Assistive technology for integration students if available	Teaching strategies	Subject			Educational activities	Formative assessment	Evaluation											
Interactive board– Textbook – Introductory Video			□ Brainstorming	□ Dialogue and discussion	□ Cooperative Learning	The concept and importance of data analysis: It is the process of examining, cleaning, and transforming data to extract useful information and make fact-based decisions. Why use Python for data analysis? It is considered one of the best programming languages for data analysis because: <table><tr><td>Easy to learn and use: Its simple structure makes it suitable for beginners.</td><td>Powerful libraries such as: Pandas for data manipulation, NumPy for scientific calculations, Matplotlib, Seaborn for graphical visualization.</td><td>A large support community with numerous educational resources.</td><td>Compatibility with artificial intelligence and machine learning</td></tr></table> A- Basic Concepts in Data Analysis: 1-Arithmetic Mean: This is the sum of the values divided by their number. Simplified code explanation: 1- import NumPy as np- (look at the student book, pages 39-40) How does the mean work? (Meaning = Sum of values / Number of values) Example: Page 40 Function: To import the NumPy library, which is used for scientific calculations in Python. Note: 'as np' is an abbreviation for the library name, allowing us to use 'np' instead of writing the full NumPy name. B- Maximum and Minimum Values: Max: The largest value in the data set. Min: The smallest value in the data set. Example: Page 41 2- Types of Data That Can Be Analyzed: <table><tr><td>A-Numerical Data: • Integers such as ages, number of products • Decimals such as weights, prices</td><td>B - Textual Data: Texts such as •customer comments, it can de be analyzed using Natural Language Processing(NLP)</td><td>C - Date and Time Data: • Dates such as purchase dates and events •can be Analyzed using pandas. to_datetime()example page 41</td></tr></table> Why use 'pd.to_datetime'() • Calculating the period between two dates. • Grouping by date, such as total monthly sales. • Extracting parts of a date (day, month, year). •Date validation: Rejects invalid dates such as “2023-13-01”. <pre>import numpy as np data = [10, 20, 30, 40, 50] mean = np.mean(data) print("المتوسط الحسابي:", mean) # 30.0: الناتج sum_data = sum(data) # 150 count = len(data) # 5 mean = sum_data / count # 30.0 print("المتوسط الحسابي:", mean)</pre>				Easy to learn and use: Its simple structure makes it suitable for beginners.	Powerful libraries such as: Pandas for data manipulation, NumPy for scientific calculations, Matplotlib, Seaborn for graphical visualization.	A large support community with numerous educational resources.	Compatibility with artificial intelligence and machine learning	A-Numerical Data: • Integers such as ages, number of products • Decimals such as weights, prices	B - Textual Data: Texts such as •customer comments, it can de be analyzed using Natural Language Processing(NLP)	C - Date and Time Data: • Dates such as purchase dates and events •can be Analyzed using pandas. to_datetime()example page 41	In collaboration with your classmates, calculate the arithmetic mean of a set of numbers.			Using PowerPoint, design a poster illustrating the most important applications of artificial intelligence.		Solve the questions in the textbook
Easy to learn and use: Its simple structure makes it suitable for beginners.	Powerful libraries such as: Pandas for data manipulation, NumPy for scientific calculations, Matplotlib, Seaborn for graphical visualization.					A large support community with numerous educational resources.	Compatibility with artificial intelligence and machine learning															
A-Numerical Data: • Integers such as ages, number of products • Decimals such as weights, prices	B - Textual Data: Texts such as •customer comments, it can de be analyzed using Natural Language Processing(NLP)					C - Date and Time Data: • Dates such as purchase dates and events •can be Analyzed using pandas. to_datetime()example page 41																
Weekly Assessment		Put a checkmark (✓) next to the correct statement and an X (x) next to the following: Data analysis helps in making fact-based decisions ()				Teacher's Signature																

Grade	Class	Date	Period	Unit	Lesson	Lesson Title
2prep.				Four	one	Unit Four: python programming language practical application of data analysis from an Excel file Lives

Introduction / What is Microsoft Excel? Think about how data is represented using Excel. Explain.

Lesson Objectives / - To learn how to import an Excel file containing student grades and view the basic data.

-To perform some calculations (calculating averages, finding the highest and lowest grades in each subject).

-To analyze the relationships between grades in different subjects using results and graphs.

Learning Objectives				Subject			Educational activities	Formative assessment	Evaluation		
Skills, values and issues covered in the lesson	Teaching Aids	Assistive technology for integration students if available	Teaching strategies								
Learn to be: Technological awareness Work values: Collaboration Globalization issues: Facing global challenges through digital transformation	Interactive board– Textbook – Introductory Video		Brainstorming □ Dialogue and discussion □ Cooperative Learning	Operating environment requirements:			In collaboration with your classmates, create a PowerPoint presentation illustrating the lesson.	Design a poster outlining the requirements for Excel	Solve the questions in the textbook.		
				<table><tr><th>The Library</th><th>The recommend libraries for (python packages)</th><th>Library function</th></tr><tr><td>pandas</td><td>pip install pandas</td><td>For analyzing data and reading Excel files.</td></tr><tr><td>matplotlib</td><td>pip install matplotlib</td><td>For creating charts and graphs.</td></tr></table> An Excel file named grades.xlsx must be located in the same folder as the program for the code to work. Look at student the data table on page 47.						The Library	The recommend libraries for (python packages)
The Library	The recommend libraries for (python packages)	Library function									
pandas	pip install pandas	For analyzing data and reading Excel files.									
matplotlib	pip install matplotlib	For creating charts and graphs.									
				First: Loading and Displaying Data: Use the following code to start the process: - import pandas as pd` Loading the Excel file:(df = pd.read_excel('grades.xlsx')` Displaying the first 5 rows. - print(df.head)` Simplified explanation of the code</li><li>import pandas as pd` - This calls the library pandas to deal with the date, tables. - pd.read_excel():This command opens the Excel file and stores it in a variable called `df`). - df.head (): This prints only the first 5 rows to ensure the data format is correct. Second: Analyzing Basic Data: 1- Extracting Useful Statistics 2- Detailing the Tools Third: Graphically Representing Data: To convert numbers into an easily understandable graph What do these commands mean? ✍ Import .matplotlib.pyplot as plt: plots the average of the materials - df.show() displays the graph. Look at the student book, page 48. "as plt" is an abbreviation for the function name. - ades.mean():- The program takes the average grades for each subject. - plot(kind='pie'): Then plots them as a pie chart. Look at S.B page 49 `kind='bar': Plots vertical bars to compare the subjects.</li><li>`kind='pie': Plots a circle showing each subject's share of the averages. `plt.show()`: The magic command that displays the graph on the screen. Fourth: Calculating the correlation between substances `print(df.corr())` 1-If the result is close to :The relationship is strong (if one substance increases, the other increases). 2-If it is close to 0: There is no clear relationship. 3-If it is close to 1: The relationship is inverse.			Teacher's Signature				
Weekly Assessment				Complete the following code: `grades.max()`							

Grade	Class	Date	Period	Unit	Lesson	Lesson Title
2prep				Four	2	conditional statements in Python Applications

Introduction / What are conditional statements? Give examples of how to use conditional statements in Python.

Lesson Objectives / -To understand the concept of conditional statements and how to use them to make decisions in programs.

-To explain how to use loops to repeat programming commands effectively.

-To apply some examples of functions and how to create and call them to organize code.

Skills, values and issues covered in the lesson				Teaching Aids	Assistive technology for integration students if available	Teaching strategies	Subject			Educational activities	Formative assessment	Evaluation			
Learn to be: Predicting the results of code execution Business values: Discovering and correcting errors Globalization issues: Addressing global challenges through digital transformation				Interactive board– Textbook – Introductory Video		☐ Brainstorming ☐ Dialogue and discussion ☐ Cooperative Learning	The Concept of Conditional Statements: In Python, conditional statements are used to make logical decisions based on the fulfillment of a specific condition. These statements tell the program to execute a specific part of the code only if the condition is true.  General Syntax for a Conditional Statement: if condition: If the condition is true execute this code else: Otherwise execute this code instead  Types of Conditional Statements in Python: 1- If statement : Example age = 18 if age >= 18 Print(“you can Enter”) 2. If.....else statement: Example Age =16 If age >=18 Print(“you can Enter”) Else: 3-if ...elif ...else statement: mark =85 if mark >=90: print(“Excellent “) elif mark =75: print(“very good”) else: print(“Failed”)  ☐ Comprehensive practical example: Code that prompts the user to enter a number and specifies its type (Positive or negative) (number = int(input " ") input number 1 if number > :o print(“positive number”) elif number < 0: print(“negative number”) else: print(“number is 0”)						In collaboration with your classmates, create a function to iterate over a given number	Compare the functions mentioned in the lesson.	Solve the questions in the textbook.
Weekly Assessment				Complete: The keyword that means "unless" in Python is				Teacher's Signature							

Grade	Class	Date	Period	Unit	Lesson	Lesson Title
2prep				Four	3	loops and functions

Introduction / What are Loops and Functions?

Lesson Objectives -To understand the concept of loops and functions.

-To explain the while loop for conditional loops.

-To write simple code for loops and functions.

Skills, values and issues covered in the lesson	Teaching Aids	Assistive technology for integration students if available	Teaching strategies	Subject	Educational activities	Formative assessment	Evaluation
Learning to be: Logical thinking and sequencing steps Work values: Explaining the concept of repetition in simple language Globalization issues: Facing global challenges through digital transformation Assigning a specific task	Interactive board– Textbook – Introductory Video		□ Brainstorming □ Dialogue and discussion □ Cooperative Learning	First: Loops : Loops help you execute a specific command multiple times without having to rewrite it over and over again . 1. `for` Loop for a known number of iterations: We use this when we know beforehand how many times we want to repeat the code <pre>for i in range(5): repeat "welcome" 5 5imes Print("welcome")</pre> 2. while` Loop for a conditional iteration: It continues to run as long as the condition is true and stops as soon as it becomes false. General syntax: While condition : Example printing numbers from 1 to 5 Pages 57 and 58 <pre>X=0 While x<3 Print("welcome") x=x+1</pre> Second: Functions :what is a function? It is as a "small machine" it is a code for Perform a specific task. Instead of writing the code 10 times, we write it once inside a function and call it whenever we want. such as calculating the sum of two numbers, printing a message, etc.).) Its benefit: It makes the code more organized and easier to understand and modify. 📌 How to create a function: ◆ The keyword `def`: This is short for `define`. We use it to create a function in Python. ◆ The function name `greet`: This is the name of the function. Any suitable name that expresses the function's purpose can be chosen. ◆ Parentheses (): These are placed after the function name. They are used later to pass values (if line) ◆ Colons: These are written immediately after the function name. the function body will begin at the next line. Indentation: Indentation is necessary for all code inside a function. Without it, an error will occur in the program. Every line inside a function must be indented. ▶ How to call a function to execute code within it: ⬇ Function with parameters ⬇ Function with input (Example p. 59) ⬇ Function that returns a result (Example p. 60) 📌 Note that: Functions make your programs faster and easier to modify. The more you practice, the more skilled a programmer you will become.	In collaboration with your classmates, create a function to iterate over a given number.	Compare the functions mentioned in the lesson.	Solve the questions in the textbook.
Weekly Assessment	Complete: The main purpose of using functions is.....				Teacher's Signature		

Grade	Class	Date	Period	Unit	Lesson	Lesson Title
2prep				Four	4	data structures or collection

Introduction / What are data structures?

Lesson Objectives / -To understand data structures in Python and why we use them.

-To compare the uses of List, Tuple, and Dictionary.

-To write simple code using data structures in Python.

Learning to be: Increasing technological awareness Work values: Mastery in technical work Globalization issues: Responsible use of technology				Subject				Educational activities		Formative assessment		Evaluation									
Interactive board– Textbook – Introductory Video																					
Teaching Aids				Assistive technology for integration students if available				Teaching strategies													
Learning to be: Increasing technological awareness Work values: Mastery in technical work Globalization issues: Responsible use of technology				What are data structures? They are methods for storing data in a program in an organized way to make it easier to work with. Examples: • Shopping List: We write the items we want to buy. • School Timetable: A fixed schedule that doesn't change throughout the week. • Glossary: A data structure that stores data as pairs of "keys" and "values".								In collaboration with your classmates, conduct research on data structures									
Interactive board– Textbook – Introductory Video				<table><thead><tr><th>Lists:</th><th>Tuple:</th><th>Dictionary</th></tr></thead><tbody><tr><td>•An editable list where items can be added or deleted, We use the square brackets [] : Example: page 63 We use it: to Add 95 : - (grades.append(95) to Delete (78) : - (grades.remove(78) to Sort the values in ascending order : - (grades.sort()</td><td>•A non-editable (static) list • Uses regular brackets. Example: page 64 We use it: •When we want static data such as: •Days of the week. • Location coordinates (latitude and longitude).</td><td>1-A key is a set of elements stored in a pair: a key-----value . 2. Keys are enclosed in curly braces {} and separated between values & crosshairs(.: 3. Keys are used for quick access to data using a key instead of an index, as in lists. 4. Keys must be unique and cannot be duplicated; they can be text or numbers. 5. New elements can be added or existing values modified easily. 6. Dictionaries are frequently used for storing structured data, such as student and employee data.</td></tr></tbody></table>				Lists:	Tuple:	Dictionary	•An editable list where items can be added or deleted, We use the square brackets [] : Example: page 63 We use it: to Add 95 : - (grades.append(95) to Delete (78) : - (grades.remove(78) to Sort the values in ascending order : - (grades.sort()	•A non-editable (static) list • Uses regular brackets. Example: page 64 We use it: •When we want static data such as: •Days of the week. • Location coordinates (latitude and longitude).	1-A key is a set of elements stored in a pair: a key-----value . 2. Keys are enclosed in curly braces {} and separated between values & crosshairs(.: 3. Keys are used for quick access to data using a key instead of an index, as in lists. 4. Keys must be unique and cannot be duplicated; they can be text or numbers. 5. New elements can be added or existing values modified easily. 6. Dictionaries are frequently used for storing structured data, such as student and employee data.	Compare different data structures.				Solve the questions in the textbook.			
Lists:	Tuple:	Dictionary																			
•An editable list where items can be added or deleted, We use the square brackets [] : Example: page 63 We use it: to Add 95 : - (grades.append(95) to Delete (78) : - (grades.remove(78) to Sort the values in ascending order : - (grades.sort()	•A non-editable (static) list • Uses regular brackets. Example: page 64 We use it: •When we want static data such as: •Days of the week. • Location coordinates (latitude and longitude).	1-A key is a set of elements stored in a pair: a key-----value . 2. Keys are enclosed in curly braces {} and separated between values & crosshairs(.: 3. Keys are used for quick access to data using a key instead of an index, as in lists. 4. Keys must be unique and cannot be duplicated; they can be text or numbers. 5. New elements can be added or existing values modified easily. 6. Dictionaries are frequently used for storing structured data, such as student and employee data.																			
Weekly Assessment				Complete: What is meant by Data Structures?				Teacher's Signature													

Grade	Class	Date	Period	Unit	Lesson	Lesson Title
2prep				Four	5	an introduction to cryptography using Python

Introduction / What is encryption? How can encryption be performed using Python?

- To perform some operations (encrypting/decrypting a single character by adding and subtracting a constant).

Learn to be: Digital Expertise Business Values: Ethical Use of Technology Globalization Issues:Raising Awareness of Data and Information Protection		Skills, values and issues covered in the lesson	Teaching Aids	Assistive technology for integration students if available	Teaching strategies	Subject			Educational activities	Formative assessment	Evaluation																
Interactive board– Textbook – Introductory Video					☐Brainstorming ☐Dialogue and iscussion ☐Cooperative Learning Encryption: It is the art of transforming information from its original, clear, and understandable form, called "Plaintext," in to another, unintelligible form called "Cipher text" so that only the person who possesses the secret key to decrypt it can read it. 1. Correct and incorrect uses of encryption 🔑 <table><tr><th>Type of Use</th><th>Example</th><th>The Explane</th></tr><tr><td>Correct (Safe) Use</td><td>Protecting messages (WhatsApp), securing passwords, secure websites from: (https://"/>)</td><td>It aims to protect and secure your messages and data from intruders</td></tr><tr><td>Incorrect (Harmful) Use:</td><td>Ransomware.</td><td>It aims to harm you by encrypting your files and demanding money</td></tr></table> Encryption Methods: Simple Encryption: Relies on fixed rules such as the shifting of each character (e.g., Caesar cipher). See page 70 of the book. Complex Encryption: Relies on mathematical equations and very long keys (used by banks and companies). Example: AES algorithm. <table><tr><td>The Concept of Encryption</td><td>Converting plain text to encrypted text using a key</td></tr><tr><td>The function ord ("A") → 65</td><td>Converting a character to its corresponding Unicode number</td></tr><tr><td>The function chr(65) → “A”</td><td>converting a number to its corresponding character</td></tr><tr><td>+ Offset Encryption</td><td>This is done by adding a fixed numerical value, the "cryptography key," to the corresponding numerical value of the original character.</td></tr><tr><td>- Decryption</td><td>The previously added constant was subtracted from the corresponding character's value to obtain the character's value before encryption.</td></tr></table>	Type of Use	Example	The Explane	Correct (Safe) Use	Protecting messages (WhatsApp), securing passwords, secure websites from: (https://"/>)	It aims to protect and secure your messages and data from intruders	Incorrect (Harmful) Use:	Ransomware.	It aims to harm you by encrypting your files and demanding money	The Concept of Encryption	Converting plain text to encrypted text using a key	The function ord ("A") → 65	Converting a character to its corresponding Unicode number	The function chr(65) → “A”	converting a number to its corresponding character	+ Offset Encryption	This is done by adding a fixed numerical value, the "cryptography key," to the corresponding numerical value of the original character.	- Decryption	The previously added constant was subtracted from the corresponding character's value to obtain the character's value before encryption.	In collaboration with your classmates, create a security code for WhatsApp.	Using PowerPoint, design a slide about encryption.	Solve the questions in the textbook.
Type of Use	Example	The Explane																									
Correct (Safe) Use	Protecting messages (WhatsApp), securing passwords, secure websites from: (https://"/>)	It aims to protect and secure your messages and data from intruders																									
Incorrect (Harmful) Use:	Ransomware.	It aims to harm you by encrypting your files and demanding money																									
The Concept of Encryption	Converting plain text to encrypted text using a key																										
The function ord ("A") → 65	Converting a character to its corresponding Unicode number																										
The function chr(65) → “A”	converting a number to its corresponding character																										
+ Offset Encryption	This is done by adding a fixed numerical value, the "cryptography key," to the corresponding numerical value of the original character.																										
- Decryption	The previously added constant was subtracted from the corresponding character's value to obtain the character's value before encryption.																										
Weekly Assessment		Complete: Simple encryption (such as the Caesar cipher) relies on..... Teacher's Signature																									

Grade	Class	Date	Period	Unit	Lesson	Lesson Title
2prep				Four	6	Python cryptography: how to turn messages into secrets using the XOR operator)

Introduction / How can you perform encryption in Python?

Lesson Objectives / -Explains the concept/function of bit-level operations.

-Uses the XOR operation to encrypt/decrypt simple messages.

-Writes a simple Python program to encrypt/decrypt text with a secret key.

Skills, values and issues covered in the lesson	Teaching Aids	Assistive technology for integration students if available	Teaching strategies	Subject	Educational activities	Formative assessment	Evaluation				
Learn to be: Logical and analytical thinking Work values: Predicting outcomes Globalization issues: Security thinking	Interactive board– Textbook – Introductory Video		□Brainstorming □Dialogue and discussion □Cooperative Learning	💡 Cryptography Basics : Bitwise Operations: These are the syntax rules of the Light language Its advantages: 1- Processing speed:The fastest way to perform arithmetic and logical operations. 2- The basic of encryption:The strongest data protection algorithms in the world are built upon it 3-The secret star: The XOR (Exclusive OR) operation. Represented in Python by the symbol ^. This process works based on the secret of the reverse operation. See the book, page 76. □ Encryption and Decryption in Python: In Python, bitwise operations cannot be directly applied to characters. Therefore, we do the following: Secret of the Reverse Operation: Helpful Tools: □`ord()` → Converts the character to its corresponding Unicode number □`chr()` → Converts the number to its corresponding Unicode character □The Constant Secret Key: In the examples, we will use: `secret key = 10` <table><tr><th>□encryption Process: Encrypting one character (A) Decryption</th><th>□ Decryption code for the previous code:</th></tr><tr><td><pre>(secret_key = 10 (original_char = "A " encrypted_char_num = (ord(original_char) ^ secret_key cipher_text = (chr(encrypted_char_num) print(f"Cipher: {cipher_text}")</pre></td><td><pre>secret_key = 10 cipher_text = "K" encrypted_char_num = ord(cipher_text original_char_num = encrypted_char_num ^ secret_key decrypted_char = chr(original_char_num) print(f"Decrypted: {decrypted_char}")</pre></td></tr></table>	□encryption Process: Encrypting one character (A) Decryption	□ Decryption code for the previous code:	<pre>(secret_key = 10 (original_char = "A " encrypted_char_num = (ord(original_char) ^ secret_key cipher_text = (chr(encrypted_char_num) print(f"Cipher: {cipher_text}")</pre>	<pre>secret_key = 10 cipher_text = "K" encrypted_char_num = ord(cipher_text original_char_num = encrypted_char_num ^ secret_key decrypted_char = chr(original_char_num) print(f"Decrypted: {decrypted_char}")</pre>	In collaboration with your classmates, decode a letter.	In collaboration with your classmates, decode a letter.	With your classmates, work on deciphering a letter.
□encryption Process: Encrypting one character (A) Decryption	□ Decryption code for the previous code:										
<pre>(secret_key = 10 (original_char = "A " encrypted_char_num = (ord(original_char) ^ secret_key cipher_text = (chr(encrypted_char_num) print(f"Cipher: {cipher_text}")</pre>	<pre>secret_key = 10 cipher_text = "K" encrypted_char_num = ord(cipher_text original_char_num = encrypted_char_num ^ secret_key decrypted_char = chr(original_char_num) print(f"Decrypted: {decrypted_char}")</pre>										
Weekly Assessment				Complete: Bit-level operations are.....	Teacher's Signature						